



Introduction to SQL Programming

Introduction

Overview

Structured Query Language (SQL) is a standard language that allows for storing, manipulating, and retrieving data from relational database management systems.

These types of databases, relational database management systems, store data in tables which are a collection of related data entries containing columns and rows.

- A record is a horizontal entry in a table while a column is a vertical entity in a table.

There are many types of relational database management systems (RDMS) some of which include:

- Oracle
- MySQL
- SQL Server
- PostgreSQL
- Microsoft Access

There are minor differences in how the above databases implement SQL, however, how they implement SQL largely remains the same. If you ever implement SQL code that doesn't work, you can always refer to the official documentation of the respective RDMS for clarity.

With SQL we can perform the following basic operations:

- We can execute queries against a database
- We can create new tables in a database
- We can retrieve data from a database
- We can insert records in a database
- We can update records in a database
- We can delete records from a database
- We can create new databases

Applications

SQL is used in almost every industry where significant amounts of data are involved.

- In the finance industry, banking applications store and operate data about financial transactions. This data is usually stored in SQL Databases i.e. Oracle Database.
- Web applications such as social media platforms like Facebook, Twitter, etc., use SQL to store user's profile information in an SQL Database i.e. Postgres or MySQL Database.
- In the health care industry, SQL is used to store patient medical records data.
- Within businesses, SQL is used for storing inventory and transactional data.

Who works with SQL?

- Database Developers
- Database Administrators
- Data Analysts
- Business Analysts
- Data Scientists

SQL Fundamentals

We will cover the following concepts as a start in becoming familiar with SQL:

- Creating a Tables
- Altering a Tables
- Dropping a Table
- Selecting Data from a Table
- Inserting a Table
- Updating Data in a Table
- Performing Calculations with SQL
- Subqueries
- Joining Tables

Creating SQL Tables

As mentioned, while working with RDMS, we use tables to store data. The CREATE TABLE statement is used to create a new table in an RDMS database.

The following example creates a table called "Students" that contains four columns: StudentID, FirstName, LastName, and Class:

```
CREATE TABLE Students (
    StudentID int,
    FirstName varchar(255),
    LastName varchar(255),
    Class varchar(255),
);
```

The StudentID column is of type int and will hold an integer.

The FirstName, LastName, and Class columns are of type varchar and will hold characters, and the maximum length for these fields is 255 characters.

Altering SQL Tables

We use The ALTER TABLE statement to add, delete, or modify columns in an existing table.

The following SQL adds an "Email" column to the "Students" table:

```
ALTER TABLE Students
ADD Email varchar(255);
```

Dropping a Table

We use the DROP TABLE statement to drop/delete an existing table in a database.

```
DROP TABLE Students;
```

Selecting Data from a Table

The SELECT statement is used to fetch records from a table.

The following (Oracle) SQL statement selects the first three records from the "Students" table:

```
SELECT * FROM Students
FETCH FIRST 5 ROWS ONLY;
```

We can also select all records from the Students table as shown below:

```
SELECT * FROM Students;
```

Inserting to a Table

The INSERT INTO statement is used to insert new records in a table.

The following SQL statement inserts a new record in the "Students" table:

```
INSERT INTO Customers (StudentID, FirstName, LastName,
Class)
VALUES ('ST001', 'Valentine', 'Mwangi', 'Data Science');
```

Updating Data in a Table

We can use the UPDATE statement is used to modify the existing records in a table.

The following SQL statement updates the student with the given ID (StudentID = ST001) with a new class.

```
UPDATE Students
SET Class= 'Frankfurt'
WHERE CustomerID = 1;
```

Performing Calculations with SQL

We can perform calculations in an SQL statement through the use of an arithmetic expression.

For example, the following SQL statement calculates the total price before discount and after discount for each order item.

```
SELECT OrderID,  
       ProductID,  
       UnitPrice*Quantity AS "Regular Price",  
       UnitPrice*Quantity-UnitPrice*Quantity*Discount AS  
       "Price After Discount"  
FROM order_details;
```

The output of this statement would be as follows:

OrderID	ProductID	Regular Price	Price After Discount
10248	11	168	168
10248	42	98	98
10248	72	174	174
10249	14	167.4	167.4
10249	51	1696	1696
10250	41	77	77
10250	51	1484	1261.3999911547
10250	65	252	214.19999849796

Source: GreeksEngine [\[Link\]](#)

Subqueries

Subqueries allow us to contain SQL statements within SQL statements. This allows us to perform queries that can join two tables as shown below.

The following subquery will list products with order quantities greater than 100. Note the two tables Product and OrderItem.

```
SELECT ProductName  
FROM Product  
WHERE Id IN (SELECT ProductId  
             FROM OrderItem  
             WHERE Quantity > 100)
```

Results: 12 records

ProductName
Guaraná Fantástica
Schoggi Schokolade
Chartreuse verte
Jack's New England Clam Chowder
Rogede sild
Manjimup Dried Apples
Perth Pasties
...

Source: DoFactory [[Link](#)]

Joining Tables

An SQL JOIN statement combines records from two tables.

In the following example, the "StudentID" column in the "Grades" table refers to the "StudentID" in the "Books" table. The relationship between the two tables is the "StudentID" column.

```
SELECT Grades.BookID, Books.FirstName, Grades.BookDate
FROM Grades
INNER JOIN Books ON Grades.StudentID=Books.StudentID;
```

Source: W3Schools [[Link](#)]

While working with SQL we should note that we can also write our SQL queries in lowercase characters.

References

- SQL Introduction to Beginners. Link: <https://learntocodewith.me/posts/sql-guide/>
- SQL. W3Schools. Link: <https://www.w3schools.com/sql/>